



The following claims are presented for examination:

1. (Currently Amended) A method comprising:
populating a cache with a resource only when at least i requests for said resource have been received;
wherein **[at least occasionally]** i is an integer **and is at least occasionally** greater than one.
2. (Original) The method of claim 1 wherein the value of i is invariant.
3. (Original) The method of claim 1 wherein the value of i is based on calendrical time.
4. (Original) The method of claim 1 wherein said cache is populated with said resource only when at least i requests for said resource have been received within an elapsed time interval, Δt .
5. (Original) The method of claim 4 wherein the duration of said elapsed time interval, Δt , is based on the value of i .
6. (Original) The method of claim 4 wherein the value of i is based on calendrical time.
7. (Original) The method of claim 4 wherein the duration of said elapsed time interval, Δt , is based on calendrical time.
8. (Currently Amended) A data processing system comprising:
a cache for storing a resource; and
a processor for populating said cache with said resource only when at least i requests for said resource have been received;
wherein i is an integer **and is at least occasionally** greater than one.
9. (Original) The data processing system of claim 8 wherein the value of i is invariant.
10. (Original) The data processing system of claim 8 wherein the value of i is based on calendrical time.
11. (Original) The data processing system of claim 8 wherein said cache is populated with said resource only when at least i requests for said resource have been received within an elapsed time interval, Δt .

12. (Original) The data processing system of claim 8 wherein the duration of said elapsed time interval, Δt , is based on the value of i .

13. (Original) The data processing system of claim 8 wherein the value of i is based on calendrical time.

14. (Original) The data processing system of claim 8 wherein the duration of said elapsed time interval, Δt , is based on calendrical time.

15. (Currently Amended) A method comprising:
receiving at a first node in a computer network at least one request for a resource;
retrieving said resource from a second node in said computer network; and
populating a cache in said first node with said resource only when at least i requests for said resource have been received at said first node;
wherein i is an integer **and is at least occasionally** greater than one.

16. (Original) The method of claim 15 wherein the value of i is invariant.

17. (Original) The method of claim 15 wherein the value of i is based on calendrical time.

18. (Original) The method of claim 15 wherein said cache is populated with said resource only when at least i requests for said resource have been received within an elapsed time interval, Δt .

19. (Original) The method of claim 18 wherein the duration of said elapsed time interval, Δt , is based on the value of i .

20. (Original) The method of claim 18 wherein the value of i is based on calendrical time.

21. (Original) The method of claim 18 wherein the duration of said elapsed time interval, Δt , is based on calendrical time.

22. (Original) The method of claim 15:
wherein said computer network is a hierarchical computer network and said first node has m filial nodes;
wherein said cache is populated with said resource only when at least one request for said resource has been received from at least n of said m filial nodes; and
wherein m is an integer greater than one, n is an integer greater than one, and $m \geq n$.

- 23.** (Original) The method of claim 15:
wherein said computer network is a hierarchical computer network and said first node has m filial nodes;
wherein said cache is populated with said resource only when at least one request for said resource has been received from at least n of said m filial nodes within an elapsed time interval, Δt ; and
wherein m is an integer greater than one, n is an integer greater than one, and $m \geq n$.
- 24.** (Currently Amended) A first node in a computer network, said first node comprising:
a cache;
at least one receiver for receiving at least one request for a resource; and
a processor for retrieving said resource from a second node in said computer network, and for populating said cache in said first node with said resource only when at least i requests for said resource have been received at said first node;
wherein i is an integer **and is at least occasionally** greater than one.
- 25.** (Original) The first node of claim 24 wherein the value of i is invariant.
- 26.** (Original) The first node of claim 24 wherein the value of i is based on calendrical time.
- 27.** (Original) The first node of claim 24 wherein said cache is populated with said resource only when at least i requests for said resource have been received within an elapsed time interval, Δt .
- 28.** (Original) The first node of claim 27 wherein the duration of said elapsed time interval, Δt , is based on the value of i .
- 29.** (Original) The first node of claim 27 wherein the value of i is based on calendrical time.
- 30.** (Original) The first node of claim 27 wherein the duration of said elapsed time interval, Δt , is based on calendrical time.
- 31.** (Original) The first node of claim 24:
wherein said computer network is a hierarchical computer network and said first node has m filial nodes;

wherein said cache is populated with said resource only when at least one request for said resource has been received from at least n of said m filial nodes; and

wherein m is an integer greater than one, n is an integer greater than one, and $m \geq n$.

32. (Original) The first node of claim 24:

wherein said computer network is a hierarchical computer network and said first node has m filial nodes;

wherein said cache is populated with said resource only when at least one request for said resource has been received from at least n of said m filial nodes within an elapsed time interval, Δt ; and

wherein m is an integer greater than one, n is an integer greater than one, and $m \geq n$.